

1/5/1 (Item 1 from file: 351)  
DIALOG(R) File 351:Derwent WPI  
(c) 2003 Thomson Derwent. All rts. reserv.

012310325 \*\*Image available\*\*  
WPI Acc No: 1999-116431/ 199910  
XRPX Acc No: N99-086020

Encoder for run length limited (RLL) symbol row used as recording symbol  
corresponding to magneto-optical disc - has control symbol inserting  
circuit which inserts displaced bit row, with which control symbol of  
selected bit pattern of run length is replaced, to RLL symbol row

Patent Assignee: SONY CORP (SONY )  
Number of Countries: 001 Number of Patents: 001  
Patent Family:

Patent No	Kind	Date	Applicat No	Kind	Date	Week
JP 10340543	A	19981222	JP 97163935	A	19970620	199910 B

Priority Applications (No Type Date): JP 9789702 A 19970408

Patent Details:

Patent No	Kind	Lan Pg	Main IPC	Filing Notes
JP 10340543	A	17	G11B-020/14	

Abstract (Basic): JP 10340543 A

NOVELTY - A control symbol selector alternately chooses the bit pattern of the control symbol of a bit length. The control symbol of the selected bit pattern is replaced with a displaced bit row. A control symbol acquisition stage inserts the displaced bit row to a RLL symbol row. DETAILED DESCRIPTION - The bit contents of the displaced bit row of a bit length are positioned before or after the insertion position of a control symbol for digital sum value (DSV). An identification circuit identifies the bit contents of predetermined symbol positioned in the displaced bit row or symbol row. The DSV control protects the limitation of the RLL symbol row based on the identified bit contents. The displaced bit row, with which the control symbol of selected bit pattern is replaced, is inserted to the RLL symbol row. INDEPENDENT CLAIMS are included for the following: decoder; encoding method; decoding method.

USE - For RLL symbol row used as recording symbol corresponding to magneto-optical disc.

ADVANTAGE - Ensures high density recording since lower redundancy is obtained when DSV control code is inserted as a redundant bit. Performs decompression of RLL symbol row before DSV control even without performing complicated process. DESCRIPTION OF DRAWING(S) - The figure shows an explanatory drawing showing the concept of the DSV control code.

Dwg.1/10

Title Terms: ENCODE; RUN; LENGTH; LIMIT; RLL; SYMBOL; ROW; RECORD; SYMBOL; CORRESPOND; MAGNETO-OPTICAL; DISC; CONTROL; SYMBOL; INSERT; CIRCUIT; INSERT; DISPLACE; BIT; ROW; CONTROL; SYMBOL; SELECT; BIT; PATTERN; RUN; LENGTH; REPLACE; RLL; SYMBOL; ROW

Derwent Class: T03; U21; W04

International Patent Class (Main): G11B-020/14

International Patent Class (Additional): G11B-011/10; H03M-007/46

File Segment: EPI

1/5/2 (Item 1 from file: 347)  
DIALOG(R) File 347:JAPIO  
(c) 2003 JPO & JAPIO. All rts. reserv.

06057443 \*\*Image available\*\*  
ENCODING DEVICE, DECODING DEVICE, ENCODING METHOD AND DECODING METHOD THEREFOR

PUB. NO.: 10-340543 A]

PUBLISHED: December 22, 1998 (19981222)  
INVENTOR(s): KAWASE KIMITAKA  
APPLICANT(s): SONY CORP [000218] (A Japanese Company or Corporation), JP  
(Japan)  
APPL. NO.: 09-163935 [JP 97163935]  
FILED: June 20, 1997 (19970620)  
INTL CLASS: [6] G11B-020/14; G11B-011/10; H03M-007/46  
JAPIO CLASS: 42.5 (ELECTRONICS -- Equipment); 42.4 (ELECTRONICS -- Basic  
Circuits)  
JAPIO KEYWORD: R138 (APPLIED ELECTRONICS -- Vertical Magnetic &  
Photomagnetic Recording)

#### ABSTRACT

PROBLEM TO BE SOLVED: To promote high density recording by keeping restriction on a set minimum run (d) and maximum run (k), inserting a digital sum value (DSV) control code by a number of bits having redundancy as less as possible and performing DSV control.

SOLUTION: In encoding processing for inserting a DSV control code into a (d, k) run length limited RLL code string for performing DSV control, the (m) bit DSV control code of a bit pattern, for keeping the (d, k) limit and performing DSV control, is substituted for the bit pattern of a bit string to be substituted having (a) bits forming the original (d, k) RLL code string and the substantial redundancy of DSV control is made to be (m-a) bits. At the time of decoding processing for acquiring the (d, k) RLL code of released DSV control, the substituted bit selected by the corresponding information of the (m) bit DSV control code bit pattern to the (a) bit substituted string bit pattern is inserted into the (d, k) RLL code string and the control code is removed.

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平10-340543

(43) 公開日 平成10年(1998)12月22日

(51) Int.Cl. <sup>a</sup>	識別記号	F I
G 1 1 B 20/14	3 4 1	G 1 1 B 20/14 3 4 1 A
11/10	5 8 6	11/10 5 8 6 F
H 0 3 M 7/46		H 0 3 M 7/46

審査請求 未請求 請求項の数 6 O L (全 17 頁)

(21) 出願番号 特願平9-163935

(22) 出願日 平成9年(1997)6月20日

(31) 優先権主張番号 特願平9-89702

(32) 優先日 平9(1997)4月8日

(33) 優先権主張国 日本 (J P)

(71) 出願人 000002185

ソニー株式会社

東京都品川区北品川6丁目7番35号

(72) 発明者 川瀬 公崇

東京都品川区北品川6丁目7番35号 ソニ

ー株式会社内

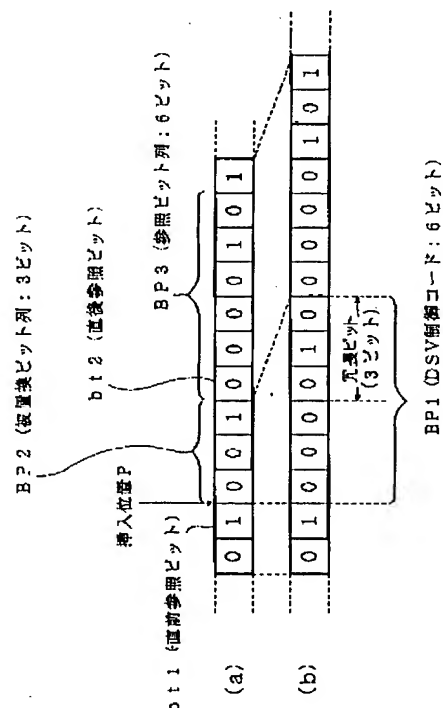
(74) 代理人 弁理士 脇 篤夫 (外1名)

(54) 【発明の名称】 エンコード装置、デコード装置、エンコード方法、及びデコード方法

(57) 【要約】

【課題】 DSV制御が行われたRLL符号として、  
(d, k) 制限を守ったうえで、できるだけ冗長度が低くなるようにする。

【解決手段】 (1, 7) RLL符号列の被置換ビット列BP2(3ビット)を、(d, k) 制限を守るための所定規則に従って選択した、6ビットのビットパターンのDSV制御コードBP1に置き換えるようにしてDSV制御を行うことで、実質的にDSV制御のために挿入される冗長ビットが3ビットで済むようにする。



## 【特許請求の範囲】

【請求項 1】 (d, k) ランレングスリミテッド符号列 (d は最小ラン数、k は最大ラン数を示す) に対してデジタルサムバリュウ制御を施すためのエンコード装置において、

上記 (d, k) ランレングスリミテッド符号列における、デジタルサムバリュウ制御のための制御符号の挿入位置の直前又は直後に位置する a ビット長の被置換ビット列のビット内容と、当該被置換ビット列の前方又は後方に位置する所定の符号又は符号列のビット内容を識別可能なビット内容識別手段と、

上記ビット内容識別手段の識別結果に基づいて、(d, k) ランレングスリミテッド符号列としての最小ラン数 d 及び最大ラン数 k の制限を守りながらデジタルサムバリュウ制御を実現することのできる、m (ただし  $m > a$ ) ビット長の制御符号のビットパターンを択一的に選択する制御符号選択手段と、

上記制御符号選択手段により選択されたビットパターンの制御符号を、上記被置換ビット列と置き換えるようにして上記 (d, k) ランレングスリミテッド符号列に対して挿入する制御符号挿入手段と、

を備えていることを特徴とするエンコード装置。

【請求項 2】 上記制御符号選択手段は、(d, k) ランレングスリミテッド符号列としての最小ラン数 d 及び最大ラン数 k の制限を守りながらデジタルサムバリュウ制御を実現することのできる制御符号のビットパターンが複数存在する場合には、これら複数のビットパターンのうち、反転制御を示すビットの数が最も少ないビットパターンを選択するように構成されていることを特徴とする請求項 1 に記載のエンコード装置。

【請求項 3】 デジタルサムバリュウ制御の解除された (d, k) ランレングスリミテッド符号列を得るためのデコード装置において、

(d, k) ランレングスリミテッド符号列に挿入されているデジタルサムバリュウ制御のための m ビット長の制御符号のビットパターンと、上記制御符号挿入前の元の (d, k) ランレングスリミテッド符号列を形成する a (ただし  $a < m$ ) ビット長の置換ビット列のビットパターンとを対応させた対応情報が格納された対応情報格納領域と、

(d, k) ランレングスリミテッド符号に挿入されている上記制御符号のビットパターンを識別する制御符号識別手段と、

上記対応情報を参照して、上記制御符号識別手段により識別した制御符号のビットパターンに対応する置換ビット列のビットパターンを選択する置換ビット列選択手段と、

上記置換ビット列選択手段により選択されたビットパターンの置換ビット列を上記制御符号と置き換えるようにして、上記 (d, k) ランレングスリミテッド符号列に

対して挿入する置換ビット挿入手段と、  
を備えていることを特徴とするデコード装置。

【請求項 4】 (d, k) ランレングスリミテッド符号列に対してデジタルサムバリュウ制御を施すためのエンコード方法において、

上記 (d, k) ランレングスリミテッド符号列における、デジタルサムバリュウ制御のための制御符号の挿入位置の直前又は直後に位置する a ビット長の被置換ビット列のビット内容と、当該被置換ビット列の前方又は後方に位置する所定の符号又は符号列のビット内容を識別するビット内容識別処理と、

上記ビット内容識別処理の識別結果に基づいて、(d, k) ランレングスリミテッド符号列としての最小ラン数 d 及び最大ラン数 k の制限を守りながらデジタルサムバリュウ制御を実現することのできる、m (ただし  $m > a$ ) ビット長の制御符号のビットパターンを択一的に選択する制御符号選択処理と、

上記制御符号選択処理により選択されたビットパターンの制御符号を、上記被置換ビット列と置き換えるようにして上記 (d, k) ランレングスリミテッド符号列に対して挿入する制御符号挿入処理と、

を実行するように構成されていることを特徴とするエンコード方法。

【請求項 5】 上記制御符号選択処理は、(d, k) ランレングスリミテッド符号列としての最小ラン数 d 及び最大ラン数 k の制限を守りながらデジタルサムバリュウ制御を実現することのできる制御符号のビットパターンが複数存在する場合には、これら複数のビットパターンのうち、反転制御を示すビットの数が最も少ないビットパターンを選択するように処理を実行することを特徴とする請求項 4 に記載のエンコード方法。

【請求項 6】 デジタルサムバリュウ制御の解除された (d, k) ランレングスリミテッド符号列を得るためのデコード方法において、

(d, k) ランレングスリミテッド符号列に挿入されたデジタルサムバリュウ制御のための m ビット長の制御符号のビットパターンと、上記制御符号挿入前の元の

(d, k) ランレングスリミテッド符号列を形成する a (ただし  $a < m$ ) ビット長の置換ビット列のビットパターンとを対応させた対応情報を所定の格納領域に格納しておき、

(d, k) ランレングスリミテッド符号に挿入されている上記制御符号のビットパターンを識別する制御符号識別処理と、

上記対応情報を参照して、上記制御符号識別手段により識別したビットパターンの制御符号に対応する置換ビット列のビットパターンを選択する置換ビット列選択処理と、

上記置換ビット列選択手段により選択されたビットパターンの置換ビット列を上記制御符号と置き換えるように

して、上記(d, k)ランレングスリミテッド符号列に対して挿入する置換ビット挿入処理と、  
を実行するように構成されていることを特徴とするデコード方法。

#### 【発明の詳細な説明】

##### 【0001】

【発明の属する技術分野】本発明は、(d, k)ランレングスリミテッド符号列に対してデジタルサムバリュ  
10 ー制御を施すためのエンコード装置／エンコード方法、及びデジタルサムバリュ  
ー制御の解除された(d, k)ランレングスリミテッド符号列を得るためのデコード装置  
／デコード方法に関する。

##### 【0002】

【従来の技術】例えばディスクメディアに適合して記録再生を行うための変調符号としてランレングスリミテ  
ッド(RLL:Run Length Limited)符号が知られている。このRLL符号は高密度記録に適していることから、  
例えば高密度光磁気ディスクに対応した記録符号として採用して好適とされる。このようなRLL符号は、  
最小ランdと最大ランkが所定となるように制限されており、設定された最小ランdと最大ランkの制限に従  
20 ってデジタル信号列を変調することにより生成される。なお、「ラン」とは「0」と「1」の2値による符号列に  
おいて、「1」と「1」との間に連続する「0」の数をいう。従って、なお、本明細書では所定の上記最小ラン  
dと最大ランkの制限に基づいて生成されたRLL符号について(d, k)RLL符号と表記する。例えば最小  
ランd=1、最大ランk=7と定められたRLL符号であれば、(1, 7)RLL符号と表記する。また、最小  
ランdと最大ランkの制限については、以降、「(d, k)制限」ということにする。

【0003】ところで、DSV(Digital Sum Value)制御を行っていないRLL符号をそのまま記録再生に用いた  
場合には、符号列の直流成分あるいは低域成分によって基準レベルのふらつきなどによる読み出しエラーや、  
記録再生時におけるサーボエラー信号の変動が起き易く、信頼性の高い記録再生動作が望めない。そこで、  
RLL符号を実際に記録符号として用いるには、DSV制御を施して、符号列の直流成分あるいは低域成分を減少  
40 させることが必要となる。

【0004】このようなRLL記録符号に対してDSV制御を行う方法としては、例えば所定ビット数のDSV  
制御コード(冗長ビット)をRLL符号列における所要の位置に対し挿入することが考えられる。このとき、挿  
入されるDSV制御コードのビットパターンは、RLL符号列における挿入位置前後のビット値の状態に応じて  
適切とされるパターンが決定される。また、この際にはRLL記録符号の(d, k)制限を守ることのできるDSV  
50 制御コードのビットパターンが決定される。そして、上記のようにしてDSV制御コードが挿入されたR

L符号列に対して例えばNRZI(Non Return to Zero Inverted)変調を行う際には、挿入されたDSV制御  
コードによって、その前後のRLL符号列の反転／非反転が制御されることになるが、これにより、RLL符号  
列のDSV値ができるだけ0となるように制御されることでDSV制御が実現される。

##### 【0005】

【発明が解決しようとする課題】ところで、高密度記録を促進するという観点によれば、データに対する冗長度  
はできるだけ低いことが要求される。従って、上記のようにして冗長ビットであるDSV制御コードを挿入する  
ことによってRLL符号に対するDSV制御を行う場合には、できるだけDSV制御コードを形成するビット長  
を短くして高密度記録化の促進を妨げないことが要求される。

##### 【0006】

【課題を解決するための手段】そこで本発明は上記した課題を解決するため、(d, k)RLL符号としての所  
定の(d, k)制限を守ったうえで、できるだけ冗長度の少ないビット数によるDSV制御コードの挿入によ  
ってDSV制御が実現されるようにして、高密度記録の促進が図られるようにすることを目的とする。

【0007】このため、(d, k)ランレングスリミテッド符号列(dは最小ラン数、kは最大ラン数を示す)  
に対してデジタルサムバリュ  
ー制御を施すためのエンコード装置において、(d, k)ランレングスリミテッド符号列におけるデジタルサムバリュ  
ー制御のための制御符号の挿入位置の直前又は直後に位置するaビット長の被置換ビット列のビット内容と、当該被置換ビット列の  
前方又は後方に位置する所定の符号又は符号列のビット内容を識別可能なビット内容識別手段と、このビット内容  
識別手段の識別結果に基づいて、(d, k)ランレングスリミテッド符号列としての最小ラン数d及び最大ラン  
30 数kの制限を守りながらデジタルサムバリュ  
ー制御を実現することのできる、m(ただしm>a)ビット長の制御符号のビットパターンを択一的に選択する制御符号  
選択手段と、この制御符号選択手段により選択されたビットパターンの制御符号を、被置換ビット列と置き換  
えるようにして(d, k)ランレングスリミテッド符号列  
40 に対して挿入する制御符号挿入手段と備えて構成することとした。

【0008】また、デジタルサムバリュ  
ー制御の解除された(d, k)ランレングスリミテッド符号列を得るためのデコード装置において、(d, k)ランレングスリ  
ミテッド符号列に挿入されているデジタルサムバリュ  
ー制御のためのmビット長の制御符号のビットパターンと、制御符号挿入前の元の(d, k)ランレングスリミ  
テッド符号列を形成するa(ただしa<m)ビット長の置換ビット列のビットパターンとを対応させた対応情報  
50 が格納された対応情報格納領域と、(d, k)ランレン

グスリミテッド符号に挿入されている制御符号のビットパターンを識別する制御符号識別手段と、対応情報を参照して、上記制御符号識別手段により識別した制御符号のビットパターンに対応する置換ビット列のビットパターンを選択する置換ビット列選択手段と、置換ビット列選択手段により選択されたビットパターンの置換ビット列を制御符号と置き換えるようにして、 $(d, k)$  ランレングスリミテッド符号列に対して挿入する置換ビット挿入手段と備えて構成することとした。

【0009】また、 $(d, k)$  ランレングスリミテッド符号列に対してデジタルサムバリュウ制御を施すためのエンコード方法において、 $(d, k)$  ランレングスリミテッド符号列における、デジタルサムバリュウ制御のための制御符号の挿入位置の直前又は直後に位置する  $a$  ビット長の被置換ビット列のビット内容と、当該被置換ビット列の前方又は後方に位置する所定の符号又は符号列のビット内容を識別するビット内容識別処理と、このビット内容識別処理の識別結果に基づいて、 $(d, k)$  ランレングスリミテッド符号列としての最小ラン数  $d$  及び最大ラン数  $k$  の制限を守りながらデジタルサムバリュウ制御を実現することのできる、 $m$  (ただし  $m > a$ ) ビット長の制御符号のビットパターンを一時的に選択する制御符号選択処理と、この制御符号選択処理により選択されたビットパターンの制御符号を被置換ビット列と置き換えるようにして  $(d, k)$  ランレングスリミテッド符号列に対して挿入する制御符号挿入処理とを実行するように構成することとした。

【0010】更に、デジタルサムバリュウ制御の解除された  $(d, k)$  ランレングスリミテッド符号列を得るためのデコード方法において、 $(d, k)$  ランレングスリミテッド符号列に挿入されたデジタルサムバリュウ制御のための  $m$  ビット長の制御符号のビットパターンと、制御符号挿入前の元の  $(d, k)$  ランレングスリミテッド符号列を形成する  $a$  (ただし  $a < m$ ) ビット長の置換ビット列のビットパターンとを対応させた対応情報を所定の格納領域に格納しておき、 $(d, k)$  ランレングスリミテッド符号に挿入されている制御符号のビットパターンを識別する制御符号識別手順と、上記対応情報を参照して、制御符号識別手順により識別したビットパターンの制御符号に対応する置換ビット列のビットパターンを選択する置換ビット列選択手順と、この置換ビット列選択手順により選択されたビットパターンの置換ビット列を制御符号と置き換えるようにして、上記  $(d, k)$  ランレングスリミテッド符号列に対して挿入する置換ビット挿入手順とを実行するように構成することとした。

【0011】上記構成によれば、DSV 制御を行うために DSV 制御コードを  $(d, k)$  RLL 符号列に対して挿入するエンコード処理時には、元の  $(d, k)$  RLL 符号列を形成する  $a$  ビットの被置換ビット列のビットパターンを、例えば  $(d, k)$  制限を守ったうえで DSV

制御を実現することのできるビットパターンを有する  $m$  ビットの DSV 制御コードに置き換えるようにされるが、これにより、DSV 制御コードとしての実質的な冗長度は  $m - a$  ビットと見なすことができる。

【0012】また、DSV 制御が解除された  $(d, k)$  RLL 符号を得るためのデコード処理時には、例えば  $m$  ビットの DSV 制御コードのビットパターンと  $a$  ビットの置換ビット列のビットパターンとを対応させた対応情報を用意し、この対応情報を参照して選択した置換ビット列のビットパターンを、DSV 制御コードと置き換えるようにして  $(d, k)$  RLL 符号列に挿入することで DSV 制御コードが除去された  $(d, k)$  RLL 符号列を得ることが可能になる。

【0013】

【発明の実施の形態】以下、図 1～図 10 を参照して本発明の実施の形態について説明する。なお、以降の説明においては、記録符号化方式として、 $(1, 7)$  RLL 変調方式及び NRZI 変調を採用しているものとして説明する。 $(1, 7)$  RLL 変調方式では最小ラン  $d = 1$ 、最大ラン  $k = 7$  が規定されている。つまり  $(d, k)$  制限として、 $'1'$  と  $'1'$  との間の  $'0'$  の数が、最小で 1 つ～最大で 7 つまでの範囲内にあるべきことが規定される。また、NRZI 変調では RLL 符号列のビット値が  $'1'$  のときにのみ符号を反転するようにして変調を行う。なお、NRZI 変調されることを前提として、 $(1, 7)$  RLL では  $'1'$  が連続しないようにされる。

【0014】また、以降の説明は次の順序で行うこととする。

(1. RLL 符号に対して行う DSV 制御の概念)

(2. 本発明に至った経緯)

(3. 本実施の形態の DSV 制御コード)

< 3-a. 本実施の形態の DSV 制御コードの概念 >

< 3-b. DSV 制御コード変換テーブル >

(4. エンコードの構成及び信号処理動作)

(5. デコードの構成及び信号処理動作)

【0015】(1. RLL 符号に対して行う DSV 制御の概念) まず、RLL 符号に対して行われる DSV 制御の基本的な概念について説明する。ここでいう DSV 制御とは、RLL 変調された符号列に対して DSV 制御コードを挿入することにより、例えば NRZI 変調後の符号状態として、符号列の DSV 値が 0 に近くなるようにデータ列の反転、非反転を制御するものである。

【0016】図 9 (a) には、RLL 変調された符号列として  $(1, 7)$  RLL 符号が示されている。ここで、図 9 (a) の符号列において矢印に示す位置が、DSV 制御コードを挿入するための位置であるとする。上記挿入位置に対して、例えば図 9 (b) に示すように、4 ビット長による DSV 制御コードを挿入したとする。この図においては、[0101] のビットパターンによる D

SV制御コードが挿入されている。そして、図9 (b) に示す (1, 7) RLL符号列についてNRZ I変調を行った場合には、図9 (c) に示す信号パターンが得られることになるが、ここでは、DSV制御コードのビットパターン内に、反転制御符号としての '1' が2つ存在するために、DSV制御コードが挿入された区間内において符号が2回反転される結果、DSV制御コードの後のデータ列は非反転制御されたことになる。つまり、DSV制御コードとしてそのビットパターン内に '1' が偶数個存在する、もしくは '1' が存在せずすべて '0' とされるものは非反転制御機能を有することになる。これに対して、DSV制御コードとしてそのビットパターン内に '1' が奇数個存在するものについては反転制御機能を有することになる。例えば図示しないが、図9 (a) に示す (1, 7) RLL符号列に対して、DSV制御コードとして、例えば [0001] のように '1' が1個 (奇数個) 存在するビットパターンを挿入した場合には、DSV制御コードの後のデータ列は図9 (b) に示す波形パターンが反転することになって、反転制御が行われることになる。

【0017】 (2. 本発明に至った経緯) 次に、上記したようなRLL符号に対するDSV制御方法を背景として、本発明に至った経緯について図10を参照して説明する。

【0018】DSV制御コードの挿入によって常に (d, k) 制限が守られるようにしてDSV制御を実現するには、例えばDSV制御コードのビットパターンとして次の条件を満たすことが必要となる。

「DSV制御コードが挿入された状態のもとで、このDSV制御コードと、その前後の符号列号との連結関係において (d, k) 制限を守ることのできるビットパターンを有すること」・・・[第1の条件]

また、上記第1の条件を満たすためには、次に記す条件が満足されていることが前提となる。

「すくなくとも、DSV制御コード自身がRLL符号としての (d, k) 制限を守ることのできるビットパターンを有していること」・・・[第2の条件]

【0019】そこで、例えば (1, 7) RLL符号に対してDSV制御を行うのに、図9の説明のようにして単に所定ビット長より成る冗長ビットとしてのDSV制御コードを挿入する方法を採った場合について考察してみる。この場合には、結果として、上記記第1の条件を常に満たすことのできるDSV制御コードとしては少なくとも4ビットが必要とされ、3ビット以下では、上記 (d, k) 制限を守ることができない場合が発生するのであるが、この根拠について図10を参照して説明する。

【0020】なお、本明細書ではRLL符号の (d, k) 制限を守ったうえでDSV制御が行われる状態を「完全DSV制御」、DSV制御のためのDSV制御コ

ードが挿入された状態ではRLL符号としての (d, k) 制限が守られない場合が発生するような状態を「不完全DSV制御」ともいうことにする。完全DSV制御の状態では、DSV制御コードとして前述した第1の条件が満たされ、不完全DSV制御の状態では第1の条件が満たされていないことになる。

【0021】図10 (a) にはDSV制御前の状態の (1, 7) RLL符号列が示されている。ここでは、図10 (a) に示す (1, 7) RLL符号列において連続しているビットA, B間に対してDSV制御コードを挿入するものとする。図10 (b) には、上記ビットA, B間に挿入し得るとされるDSV制御コードの候補として、ビットA, Bのビット値及び反転/非反転のDSV制御種別の条件ごとに、少なくとも第2の条件を満足しているDSV制御コードのビットパターンが示されている。また、図10 (b) ではDSV制御コードのビット長として2ビット、3ビット、及び4ビットの場合がそれぞれ示されている。以下、DSV制御コードが2ビット、3ビット、4ビットの場合についてそれぞれ説明する。

【0022】ここで、(1, 7) RLLの変調規則に従った場合には、ビット (A, B) がとり得る値の組合せは、(0, 0)、(1, 0)、(0, 1) のうちのいずれかとなる。2ビットによるDSV制御コードを図10 (a) に示すビットA, B間に挿入する場合として、ビット (A, B) の値が (0, 0) である場合に、DSV制御種別として反転制御を行うためには、第2の条件を満足するDSV制御コードのビットパターンとしては、図10 (b) に示すように、[10] もしくは [01] が候補となる。例えばこの場合には、ビットA以前、もしくは、ビットB以降のビット列の状態として、最大で [000000] のように '0' が6回連続している可能性があるが、例えばビットA以前のビット列において '0' が6回連続しているとしても、DSV制御コードのビットパターンとして上記 [10] を選択して挿入することで、'0' の連続回数は6回となって (1, 7) RLLとしての (d, k) 制限を守ることができる。同様にビットB以降のビット列において '0' が6回連続しているときには、DSV制御コードのビットパターンとして [01] を選択すれば第1の条件を満足する。このように、上記2つのビットパターン候補を適宜選択して挿入することで第1の条件が満足されて完全DSV制御を行うことが可能となる。

【0023】これに対して、非反転制御を行うためには、第2の条件を満足するDSV制御コードのビットパターンとしては [00] しか候補となり得ない。この場合、ビットA以前、もしくはビットB以降のビット列において '0' が6回連続している状態であると、挿入されたDSV制御コードを含めて '0' が8回以上連続することを避けられず、従って第1の条件を満たすことが

できずに不完全DSV制御となる。

【0024】以下、同様に図10(b)に示すように、ビット(A, B)の値が(1, 0)の場合において反転制御を行うためにはDSV制御コードのビットパターンとして[01]を選択することで完全DSV制御が可能となるのに対して、非反転制御を行うためにはDSV制御コードのビットパターンとして第2の条件を守るには[00]しか候補となり得ないため、ビットB以降のビット列において'0'が6回以上連続している状態では第1の条件が満たされずに不完全DSV制御となる。同様に、ビット(A, B)の値が(0, 1)の場合においても、反転制御を行うためにはDSV制御コードのビットパターンとして[10]を選択することで完全DSV制御が可能となるのに対して、非反転制御を行うためには、第2の条件を満足するには[00]しか候補となり得ないため、ビットA以前のビット列において'0'が6回以上連続している状態では不完全DSV制御となる。

【0025】このように、DSV制御コードが2ビットである場合、非反転制御を行うためには、第2の条件を満足するDSV制御コードのビットパターンとして[00]しか候補となり得ない。このために、DSV制御コードが挿入されたRL L符号列においては、'0'が8回以上連続して(d, k)制限が守られない可能性があることから、完全DSV制御は行われなことが理解される。

【0026】次にDSV制御コードが3ビットの場合には次のようになる。まず、ビット(A, B)の値が

(0, 0)のときに反転制御を行うためには、第2の条件を満足するDSV制御コードのビットパターンとして[010]が候補となり、この場合には、ビットA以前、もしくはビットB以降のビット列において'0'が6回連続している状態であっても完全DSV制御が可能となる。また、非反転制御を行うときには、DSV制御コードのビットパターンとして[101]とすることで第1の条件が満たされて完全DSV制御が可能となる。

【0027】また、ビット(A, B)の値が(1, 0)のときに反転制御を行うためには、第2の条件を満足するDSV制御コードのビットパターンとして[001]を選択することで、ビットB以降のビット列において'0'が7回連続している状態であっても、第1の条件は満たされ、完全DSV制御が可能となる。これに対して、非反転制御を行うためには、第2の条件を満たすDSV制御コードのビットパターンの候補は[000]のみとなるが、ビットB以降のビット列において'0'が5回以上連続したときには(d, k)制限を守ることができず、つまり第1の条件が満たされずに不完全DSV制御となる。同様に、ビット(A, B)の値が(0, 1)の場合、反転制御を行うには、DSV制御コードのビットパターンとして[100]を用いれば、完全DS

V制御が可能となるのに対して、非反転制御を行うときには、第2の条件を満たすDSV制御コードのビットパターンの候補は[000]のみとなり、やはり、ビットA以前のビット列において'0'が5回以上連続したときには(d, k)制限を守ることができずに不完全DSV制御となる。このように、DSV制御コードが3ビットの場合も完全DSV制御は実現されないことになる。

【0028】これに対してDSV制御コードが4ビットの場合には次のようになる。まず、ビット(A, B)の値が(0, 0)の場合に反転制御を行うためには、第2の条件を満足するDSV制御コードのビットパターンとして[0100]又は[0010]が候補となる。このビットパターンであれば、ビットA以前、もしくはビットB以降のビット列において'0'が6回連続している状態であっても、上記[0100]又は[0010]のビットパターンのいずれかをを用いることで第1の条件は満たされ、完全DSV制御が可能となる。また、非反転制御を行うためには、DSV制御コードのビットパターンとして[1001]を挿入することで第1の条件が満たされて完全DSV制御が可能となる。また、ビット(A, B)の値が(1, 0)の場合に反転制御を行うためには、DSV制御コードのビットパターンとして[0001]を挿入することで、完全DSV制御が可能となり、非反転制御を行うためには[0101]を挿入すれば完全DSV制御が可能となる。さらに、ビット(A, B)の値が(0, 1)の場合に反転制御を行うためには、DSV制御コードのビットパターンとして[1000]を挿入することで、完全DSV制御が可能となり、非反転制御を行うためには[1010]を挿入すれば完全DSV制御が可能となる。このように、DSV制御コードが4ビットとされた場合には常に第1の条件を満たす完全DSV制御が可能となる。

【0029】これまでの説明から分かるように、(1, 7)RL L符号列に対して単に冗長ビットとしてのDSV制御コードを挿入するようにして常にDSV制御を行おうとした場合には、DSV制御コードは最小で4ビットが必要となることが理解される。なお、上記説明では、(1, 7)RL L符号列の場合を例に挙げたが、例えば他の(d, k)RL L符号列においても、上記図10による説明に準じてそれぞれ固有の(d, k)制限に応じて完全DSV制御が可能なDSV制御コードの最小ビット数が求められることになる。この完全DSV制御が可能なDSV制御コードの最小ビット数 $B_{min}$ は、 $B_{min} = 2 \times (d + 1)$ により与えることができる。

【0030】ここで、(d, k)RL L符号列に対するDSV制御コードの挿入間隔が同一の条件であるとすれば、(d, k)RL L符号列に対して、完全DSV制御が可能とされる最小ビット数((1, 7)RL L符号の場合であれば4ビット)のDSV制御コードを挿入した



場合が最も低い冗長度を有することのできる限度となるのであるが、DSV制御コードは、(d, k) RLL符号列に対して所定区間ごとに挿入されるものである。このため、DSV制御を行うのに、完全DSVが可能な最小ビットによるDSV制御コードを単に冗長ビットとして挿入するようにしたとしても、(d, k) RLL符号列全体としての冗長度は相当なものとなるため、例えば高密度記録の観点からは不利となる。

【0031】そこで本発明では、(d, k) RLL符号列に対して挿入すべきDSV制御のための冗長ビットとして、上記完全DSV制御が可能な最小ビット数よりも少ないビット長が得られるようにして、DSV制御が行われた(d, k) RLL符号として更に低い冗長度が得られるように構成するものである。

【0032】(3. 本実施の形態のDSV制御コード) <3-a. 本実施の形態のDSV制御コードの概念>図1(a)には、DSV制御前の(1, 7) RLL符号列が示されている。本実施の形態においては、図1(a)に示す(1, 7) RLL符号列におけるDSV制御コードの挿入位置Pの直後3ビットを被置換ビット列BP2として定義する。そして、DSV制御時には、この被置換ビット列BP2を、後述する変換テーブルに基づいて所要のビットパターンを有する6ビットのDSV制御コードBP1に変換し、図1(b)に示すように、上記DSV制御コードBP1を被置換ビット列BP2と置き換えるようにして(1, 7) RLL符号列に対して挿入する。このようなDSV制御コードの挿入方法によれば、DSV制御のために挿入される冗長ビットは3ビットとなる。これにより、例えば先に図9に示したようにして4ビットの冗長ビット列をDSV制御コードとして挿入する場合よりも、DSV制御が行われた(1, 7) RLL符号としての冗長度を低くすることが可能となる。

【0033】なお、以降の説明にあたり、図1(a)に示す(1, 7) RLL符号列において、挿入位置Pの直前に位置するビットを直前参照ビットbt1といい、挿入位置Pの直後に位置するビットを直後参照ビットbt2といい、更に、挿入位置Pの直後に位置する6ビットにより形成されるビット列を参照ビット列BP3ということにする。これら直前参照ビットbt1、直後参照ビットbt2、参照ビット列BP3は、後述するようにして被置換ビット列BP2のビットパターンに対応して選択されるべきDSV制御コードのビットパターンを決定するにあたり、必要に応じてそのビット内容が参照される。

【0034】<3-b. DSV制御コード変換テーブル>次に、本実施の形態の変換テーブルについて、上記図1及び、図2～図4を参照して説明する。この変換テーブルは、被置換ビット列BP2のビットパターンに対するDSV制御コードBP1のビットパターンの対応を示

すテーブルであり、このテーブルを参照することによって、置換の対象となる被置換ビット列BP2に置き換わるDSV制御コードBP1のビットパターンを得ることができる。

【0035】図2は、本実施の形態のDSV制御コードの候補を示すものであり、ここでは前述した第2の条件を満足することのできるDSV制御コードが候補として挙げられている。つまり、DSV制御コード自体が

(1, 7) RLL符号としての(d, k)制限を守ることのできるビットパターンを有するものである。このような第2の条件を満足するビットパターンを有するDSV制御コードとしては、図2に示すように、No. 1の[000000]～No. 21の[101010]の21パターンが候補となることがわかる。

【0036】図3は、本実施の形態の変換テーブルの内容を模式的に示している。以下、この変換テーブルの内容について説明を行っていく。(1, 7) RLL符号列における図1に示した3ビットの被置換ビット列BP2としては、(1, 7) RLL符号の(d, k)制限に従って、図3における最も左の列に示すように、[000], [001], [010], [100], [101]のうち何れかのビットパターンを取り得ることになる。また、左から2番目の列にはDSV制御コードのビットパターンが示されており、これらのビットパターンは図2に示したNo. 1～No. 21の候補の中から選択されている。左から3番目の列には非反転/反転の制御種別が示されている。残る最も右の列には、後述する直前参照ビットbt1、直後参照ビットbt2及び参照ビット列BP3のうちの何れかの所要のビット内容の条件が示されている。

【0037】先ず、被置換ビット列BP2が[000]の場合について考えた場合、DSV制御前の(1, 7) RLL符号列における挿入位置Pの直前あるいは直後のビット内容として、[0000]のパターンが発生する可能性があることになる。また、挿入位置Pの直前参照ビットbt1あるいは直後参照ビットbt2が‘1’となる可能性がある。従って、非反転制御のときにはDSV制御コードとして、例えば、図2に示した候補の中から[001010]のビットパターンを選択し、このビットパターンを置換して挿入すれば、DSV制御コードBP1挿入後の(1, 7) RLL符号列においては‘0’の最大連続数が7以内に納まるため、第1の条件を満たして完全DSV制御を行うことが可能になる。また、反転制御のときには[000100]のビットパターンを挿入すれば、同様にDSV制御コードBP1挿入後の(1, 7) RLL符号列における‘0’の最大連続数を7以内とすることができて第1の条件を満たした完全DSV制御が可能となる。従って、図2に示すように、被置換ビット列BP2が[000]の場合に、DSV制御コードは非反転制御の時には[001010]を

対応させ、反転制御の時には[000100]を対応させるものとする。

【0038】被置換ビット列BP2が[001]の場合は、(1, 7)RL符号列における挿入位置Pの直前ビット列では[00000]のパターンが発生し、直後のビット列においては[0000000]のパターンが発生する可能性がある。また、直前参照ビットbt1が‘1’である可能性がある。そこで、非反転制御のときには、DSV制御コードとして[001001]のビットパターンを選択すれば第1の条件を満たすことが可能となる。また、反転制御を行うときには、図1(a)に示される直前参照ビットbt1及び参照ビット列BP3のビット状態に応じて、DSV制御コードとして2種類のビットパターンの何れか一方を選択する。ここで、被置換ビット列BP2[001]の場合の反転制御時において、上記直前参照ビットbt1が‘1’で、かつ直後参照ビット内において1つ以上‘1’が存在する場合には、例えばDSV制御コードとして[000010]のビットパターンを選択すれば、DSV制御コード挿入後の(1, 7)RL符号列においては‘0’が7以上連続することがなくなり、第1の条件を満たすことが可能となる。そして、直前参照ビットbt1及び参照ビット列BP3のビット状態が上記以外の場合には、DSV制御コードとして[010101]を選択することによって第1の条件を満たすようにされる。

【0039】ところで、上記被置換ビット列BP2が[001]の場合においては、DSV制御コードとして[010101]のビットパターンのみを選択しても完全DSV制御が可能となるのであるが、直前参照ビットbt1及び参照ビット列BP3が上記のようなビット状態であるときに[000010]のビットパターンを選択するのは、次のような理由による。

【0040】図4(a)には、被置換ビット列BP2を含むDSV制御前の(1, 7)RL符号列が示されているが、仮に、このビット列に対して、[010101]のDSV制御コードを置き換え挿入することによってDSV制御を行ったとする。DSV制御後の(1, 7)RL符号列の状態としては図4(b)に示すようになるが、図4(b)に示す符号列に対してNRZI変調を施した場合には、図4(c)に示す波形の符号が得られることになる。図4(c)に示すNRZI変調符号では、[010101]という‘0’と‘1’が交互となるようなビットパターンに対して変調をかけたことにより、区間Aにおいて2Tの最小反転区間が連続して反転する現象が発生している。上記区間Aのように最小反転区間が連続する現象は、例えば波形干渉等によるジッターを発生させる原因となる。このため、最小反転区間が連続するような波形パターンはできるだけ避けることが好ましい。

【0041】そこで、本実施の形態のように、被置換ビ

ット列BP2の前後のビット状態を参照して、可能な場合には、例えば上記[000010]のように‘0’と‘1’が交互に連続しない、つまり、できるだけ反転制御符号である‘1’が少ないDSV制御コードのビットパターンを選択することで、2Tの最小反転区間が連続しないようにしている。例えば、図4(a)に示した(1, 7)RL符号列に対して、図4(d)に示すようにDSV制御コード[000010]を置き換え挿入した場合には、NRZI変調符号は図4(e)に示す波形となるが、図4(e)に示す波形では最小反転区間は連続していないことが分かる。

【0042】図3にもどり、被置換ビット列BP2が[010]の場合について説明する。この場合には、(1, 7)RL符号列における挿入位置Pの前後においては、[000000]のパターンが発生する可能性がある。従って、非反転時においては、DSV制御コードのビットパターンとして[010010]を選択すれば、置き換え挿入後の(1, 7)RL符号列においては、‘0’の連続回数が7以内となって完全DSV制御が可能である。

【0043】これに対して、非反転制御時においては、直前参照ビットbt1と直後参照ビットbt2を参照し、そのビット状態に応じて3パターンのDSV制御コードを選択する。なお、直後参照ビットbt2は、図1(a)に示すように被置換ビット列BP2の直後に位置するビットである。まず、直前参照ビットbt1が‘0’で、かつ直後参照ビットbt2が‘0’の場合には、被置換ビット列BP2の前後に[000000]のパターンが発生する可能性があることから、ここではDSV制御コードとして[100101]を選択する。これにより、DSV制御コードの置き換え挿入後の(1, 7)RL符号列の前後においては、‘0’の連続が6回以内となって第1の条件を満たすことができる。

【0044】続いて、直前参照ビットbt1が‘1’で、かつ直後参照ビットbt2が‘0’の場合には、挿入位置Pの後ろに[000000]のパターンが発生する可能性があることになる。従って、ここではDSV制御コードとして[000001]のビットパターンを選択して第1の条件を満たすようにする。更に、直後参照ビットbt2が‘1’である場合には、最も好ましいDSV制御コードビットパターンとして[010000]を選択する。これにより、第1の条件を満足するDSV制御を行うことが可能となる。

【0045】被置換ビット列BP2が[100]の場合には、(1, 7)RL符号列における挿入位置Pの直前において[0000000]のパターンが発生する可能性があり、直後においては[00000]が発生する可能性がある。そこで、非反転制御時にはDSV制御コードとして[100100]のビットパターンを選択することで第1の条件を満足するようにする。これに対し

て非反転制御時においては、直後参照ビットb t 2が  
 ‘1’のときと‘0’のときとで異なるビットパターン  
 を選択する。まず、直後参照ビットb t 2が‘1’の  
 ときには、[100000]のビットパターンを選択する  
 ことにより、仮に挿入位置直前が[0000000]で  
 あっても第1の条件が守られると共に最小反転区間の連  
 続回数も抑えることができる。この場合には、例えばD  
 S V制御コードとして[101010]のビットパター  
 ンを選択することもできるが、図4にて説明したように  
 [101010]の選択を避けることで最小反転区間の  
 連続回数を抑えるようにしている。また、直後参照ビッ  
 トb t 2が‘0’のときには[101010]を選択す  
 ることにより、第1の条件を満足するようにする。

【0046】被置換ビット列B P 2が[101]の場合  
 には、(1, 7) R L L符号列における挿入位置Pの前  
 後において共に[0000000]のパターンが発生す  
 る可能性がある。従って、非反転制御時にはD S V制御  
 コードとして[100001]を選択することで、第1  
 の条件を満足すると共に最小反転区間の連続回数を抑え  
 るようにすることができる。また、反転制御時において  
 D S V制御コードとして[101001]を選択する  
 ことにより、第1の条件を満足すると共に最小反転区間  
 の連続回数を抑えるようにされる。

【0047】本実施の形態では、上記図3に示した被置  
 換ビット列B P 2のビットパターンに対するD S V制御  
 コードB P 1のビットパターンの対応内容を変換テーブ  
 ル5 aとして用意する。そして、この変換テーブル5 a  
 の内容に基づいて、逐次、(1, 7) R L L符号列にお  
 ける被置換ビット列B P 2に対して、適切なビットパ  
 ターンのD S V制御コードを置き換え挿入するようにして  
 D S V制御を行う。

【0048】(4. エンコーダの構成及び信号処理動  
 作) 次に、図5及び図6を参照して、本実施の形態とし  
 てのエンコーダ及びその信号処理動作について説明す  
 る。図5は、本実施の形態のエンコーダの構成例を示す  
 ブロック図である。この図に示すエンコーダは、入力デ  
 ータに対して(1, 7) R L L変調処理、D S V制御、  
 及びN R Z I変調のための信号処理が可能のように構成  
 されている。

【0049】図5においては、入力データD 1が(1,  
 7) R L Lエンコーダ1に対して入力される。(1,  
 7) R L Lエンコーダ1では、入力データD 1について  
 (1, 7) R L L変調を施し、(1, 7) R L L符号D  
 2としてシフトレジスタ2に対して供給する。シフトレ  
 ジスタ2では、入力された(1, 7) R L L符号D 2に  
 ついて所要のタイミングでシフトを行う。そして、この  
 入力された(1, 7) R L L符号D 2のデータ列のうち  
 から、被置換ビット列B P 2、直前参照ビットb t 1及  
 び参照ビット列B P 3(直後参照ビットb t 2を含む)  
 の各データをデータD 6としてD S V制御コード選択回

路5に供給する。また、D S V演算を行うのに必要とな  
 るD S V制御コード挿入位置P前後の所定ビット長のデ  
 ータ列をD S V演算データD 7としてD S V演算器6に  
 供給する。また、後述するセレクト3における6ビットの  
 D S V制御コードの挿入タイミングに対応するように  
 (1, 7) R L L符号D 2についてシフトを行って、  
 (1, 7) R L L符号D 3としてセレクト3に出力す  
 る。

【0050】D S V制御コード選択回路5においては、  
 図3に示す内容の対応情報が格納された変換テーブル5  
 aが備えられている。このD S V制御コード選択回路5  
 に対しては、前述のように、シフトレジスタ2からデー  
 タD 6として、被置換ビット列B P 2、直前参照ビット  
 b t 1、及び参照ビット列B P 3(直後参照ビットb t  
 2を含む)の情報が入力される。そこで、D S V制御コ  
 ード選択回路5では、入力された被置換ビット列B P 2  
 自体のビットパターンと、必要な場合には、直前参照ビ  
 ットb t 1、参照ビット列B P 3、直後参照ビットb t  
 2のうちの所要の情報を上記変換テーブル5 aの内容と  
 照らし合わせる。これにより、入力された被置換ビッ  
 ト列B P 2に対応する非反転制御用と反転制御用の2つの  
 D S V制御コードを選択して、これら選択されたD S V  
 制御コードをD S V演算器6に対してデータD 8として  
 出力する。

【0051】タイミングジェネレータ7では、例えば  
 (1, 7) R L Lエンコーダ1から入力された符号列に  
 基づいて、D S V演算器6及びセレクト3における処理  
 動作に必要なとされるタイミング信号D 9、D 11を生成  
 して出力する。

【0052】D S V演算器6においては、データD 7と  
 して入力されたD S V制御コード挿入位置P前後のデー  
 タ列に対して、データD 8として入力された非反転制御  
 用と反転制御用のD S V制御コードをそれぞれ挿入する  
 ことにより、非反転制御用D S V制御コードが挿入され  
 たD S V演算データD aと、反転制御用D S V制御コ  
 ードが挿入されたD S V演算データD bの2つのD S V演  
 算データブロックを生成する。そして、これら2つのD  
 S V演算データブロックに基づいてD S V値の演  
 算を行い、D S V値が小さい方のD S V演算データブロ  
 ックに挿入されていたD S V制御コードを、実際に  
 (1, 7) R L L符号に挿入すべきD S V制御コードD  
 10として選択して、セレクト3に出力する。上記のよ  
 うなD S V演算器6における演算比較処理は、タイミン  
 グジェネレータ7から供給されるタイミング信号D 9に  
 基づいたタイミングにより実行される。

【0053】セレクト3は、シフトレジスタ2から供給  
 された(1, 7) R L L符号D 3に対して、(1, 7)  
 R L L符号D 3における被置換ビット列B P 2をD S V  
 制御コードD 10に置き換え挿入することにより、D S  
 V制御が行われた(1, 7) R L L符号列D 4を生成し

て出力する。上記(1, 7) RLL符号列D4は、NRZ1エンコード4に供給されてNRZ1変調処理が施された後に、記録データとして出力される。この記録データは、例えば所定種類のディスク状記録媒体に対して記録するための記録データとして利用されることになる。

【0054】図6は、上記図5に示した構成によるデコーダの信号処理動作例を示すタイミングチャートであり、図5と同一のデータ、信号については同一符号を付している。例えば、図6(a)に示すタイミング及びビット列による入力データD1が(1, 7) RLLエンコード1に入力されたとすると、(1, 7) RLLエンコード1はこの入力データ列に対して(1, 7) RLL変調を施して、図6(b)に示すビット列による(1, 7) RLL符号D2を生成してシフトレジスタ2に出力する。シフトレジスタ2では、図6(b)に示す(1, 7) RLL符号D2のうち、被置換ビット列BP2、直前参照ビットb t1及び参照ビット列BP3(直後参照ビットb t2を含む)をデータD6としてビットパターン選択回路5に供給する。また、この場合には、例えば図6(b)に示す(1, 7) RLL符号D2における被置換ビット列BP2の前後のビット列を、DSV演算データD7としてDSV演算器6に供給する。この図では、置換ビット列BP2の前の9ビットによる「010010100」と、後ろの9ビットによる「100010100」がDSV演算データD7となる。また、図6(b)に示す(1, 7) RLL符号D2を、例えば図6(h)に示すタイミングとなるようにシフトして、(1, 7) RLL符号D3としてセクタ3に出力する。図6(h)に示すデータD3のシフトタイミングは、例えばDSV演算6の処理時間に対応して設定される。

【0055】図6においては、被置換ビット列BP2のビットパターンは「000」である。そこで、ビットパターン選択回路5においては、内部に保持している変換テーブルの内容を参照して、被置換ビット列BP2「000」のビットパターンが対応するDSV制御コードのビットパターンを選択する。図3の変換テーブルによれば、この場合にはDSV制御コードのビットパターンとして、非反転制御のための「001010」と反転制御のための「000100」が選択されることになる。

【0056】なお、上記のように被置換ビット列BP2のビットパターンが「000」である場合には、直前参照ビットb t1、参照ビット列BP3、直後参照ビットb t2は選択のための条件として利用しないで済むのであるが、例えば前述のように、被置換ビット列BP2が「001」、「010」、あるいは「100」の場合には、直前参照ビットb t1、参照ビット列BP3、直後参照ビットb t2のうちから必要に応じた条件を参照して反転制御のためのDSV制御コードのビットパターンを選択することになる。

【0057】上述のようにしてビットパターン選択回路5によって選択されたDSV制御コードのビットパターン「001010」(非反転制御)と、「000100」(反転制御)は、データD8としてDSV演算器6に入力される。この場合、DSV演算器6では、上記データD8と、シフトレジスタ2から供給されたDSV演算データD7とに基づいて、図6(c)に示すDSV演算ブロックDa及び図6(e)に示すDSV演算ブロックDcを生成する。DSV演算ブロックDaは、DSV演算データD7に対して非反転制御のためのDSV制御コード「001010」を挿入して生成したものであり、DSV演算ブロックDcは、DSV演算データD7に対して反転制御のためのDSV制御コード「000100」を挿入して生成したものである。ここで、上記DSV演算ブロックDa、Dcのデータ列に対してNRZ1制御をかけた場合には、それぞれ図6(d)に示すNRZ1符号列Dbと、図6(f)に示すNRZ1符号列Ddが得られることになる。DSV演算器6では、このようにして得られるデータに基づいてDSV値を算出する。

【0058】DSV演算器6におけるDSV値の算出方法は、例えば図6(g)に示すように行われる。先ず、現在まで累積されたDSV値に対して、新たに供給されたDSV演算データD7(ここではビット列「010010100」となる)による値を加算する。なお、DSV値の演算は、図6(d)、(f)に示すNRZ1変調時の符号列の極性に基づいて加算を行うものであり、ここではNRZ1変調符号列がHレベル('1')の時に加算を行い、Lレベル('0')の時に減算を行うものとされる。続いて、このDSV値の加算結果に対して、非反転制御のDSV制御コードのビットパターン(「001010」)による値を加算したDSV値と、反転制御のDSV制御コードのビットパターン(「000100」)による値を加算したDSV値との2通りのDSV値を算出する。さらに、これら2通りのDSV値に対して、次に供給されたDSV演算データD7(「100010100」)による値を加算してDSV値を得る。

【0059】このようにしてDSV値を算出した結果、この場合には、図6(g)に示す非反転制御のDSV演算ブロックDaによるDSV値(実線)と、反転制御のDSV演算ブロックDcによるDSV値(波線)が得られることになる。この演算結果によると、非反転制御のDSV演算ブロックDaによるDSV値の方が小さな値となることが判断される。そこでこの場合、DSV演算器6は、図6(b)に示される被置換ビット列BP2

「000」に置き換え挿入すべきDSV制御コードとして、非反転制御のためのビットパターン「001010」を選択する。そして、このビットパターン「001010」をDSV制御コードD10としてセクタ3に

出力する。なお、上記したDSV演算器6の動作は、図6(i)に示すタイミング信号D9に基づくタイミングにより行われるものとされる。

【0060】セクタ3には、前述のように図6(h)に示す(1, 7)RLL符号D3と、図6(j)に示すDSV制御コードD10と、図6(k)に示すタイミング信号D11が入力される。そして、セクタ3は、図6(k)のタイミング信号D11がHレベルの期間においては、図6(h)に示す(1, 7)RLL符号D3を

選択してそのまま出力し、タイミング信号D11がLレベルの期間においては、DSV制御コードD10を選択して出力するように動作する。ここで、タイミング信号D11はタイミングジェネレータ7において、図6

(h)(j)(k)により示されるDSV制御コードD10の挿入タイミングが実現されるように生成される信号である。これにより、セクタ3から出力される(1, 7)RLL符号列D4としては、図6(1)に示すように、被置換ビット列BP2がDSV制御コードD10に置き換わるようにして挿入されたものとなる。このようにしてDSV制御が行われた(1, 7)RLL符号列D4では、DSV制御コードD10を含むその前後において前述した第1の条件が守られることにより、完全DSV制御が実現されていることになる。図6(m)には、NRZIエンコーダにより上記(1, 7)RLL符号列D4についてNRZI変調を施した符号列の波形が示されている。

【0061】(5. デコードの構成及び信号処理動作)次に、本実施の形態によるDSV制御が行われた(1, 7)RLL符号列からDSV制御コードを除去して、もとの(1, 7)RLL符号列に復元することのできるデコードの構成について、図7及び図8を参照して説明する。図7は本実施の形態としてのデコードの回路構成を示すブロック図であり、図8はデコードの信号処理動作を示すタイミングチャートである。

【0062】例えば、所定種類のディスク状記録媒体に対する再生動作により読み出された再生データD100は、図7に示すNRZIデコーダ11に対して供給される。NRZIデコーダ11は、例えば入力された再生データD100を1ビット遅延させた遅延データD101を出力する遅延器10aと、遅延データD101と再生データD100について加算を行うことによりNRZI復調された(1, 7)RLL符号列D102を出力する加算器10bを備えて構成される。なお、加算器10bでは、実際には遅延データD101と再生データD100のEX-ORをとるように構成されている。例えば、上記再生データD100が図8(a)に示す波形であった場合、遅延器10aにより再生データD100を1ビット遅延させた遅延データD101の波形は図8(b)に示され、また、NRZIデコーダ11によりNRZI復調される(1, 7)RLL符号列D102(加算器1

0bの演算出力)は図8(c)に示される状態の符号列となる。この(1, 7)RLL符号列D102は、シフトレジスタ12に出力されると共に、例えばタイミングジェネレータ16に対して入力されて、後述するタイミング信号D108、D109を発生させるために用いられる。

【0063】シフトレジスタ12は、後述するコードパターン選択回路15の動作に適合して設定された所定タイミングにより(1, 7)RLL符号列D102をシフトし、(1, 7)RLL符号列D103としてセクタ13に出力する。この場合、(1, 7)RLL符号列D103は、図8(c)に示す(1, 7)RLL符号列D102に対して、図8(f)に示すタイミングにまでシフトされて出力されるものとする。また、シフトレジスタ12では、入力された(1, 7)RLL符号列D102に含まれるDSV制御コード(BP1)を抜き出してDSV制御コードD106としてビットパターン選択回路15に供給する。この場合には、図8(c)に示す

(1, 7)RLL符号列D102において[001010]のビットパターンによるDSV制御コードD106がビットパターン選択回路15に供給される。

【0064】ビットパターン選択回路15は、図3に示したと同様の内容の変換テーブル15aを備えているものとされる。ただし、デコード時においては単にDSV制御コードを後述する置換ビット列BP2(エンコード時の被置換ビット列BP2と同等であるため、同一符号「BP2」を付すこととする)に置き換え挿入すればよく、エンコード時のように、反転時と非反転時の区別は不要であり、また、挿入位置前後の所定ビットあるいは所定ビット列のビット状態を参照する必要もない。従って、ビットパターン選択回路15の変換テーブルの内容としては、DSV制御コードのビットパターンに対する置換ビット列のビットパターンの対応が示されていればよい。つまり、図3における最も左の列に示される置換ビット列のビットパターンと、その左隣の列のDSV制御コードのビットパターンの内容が保持されていればよい。

【0065】そして、ビットパターン選択回路15では、変換テーブル15aを参照して、DSV制御コード[001010]に対応する置換ビット列BP2を選択する。図3によれば、DSV制御コード[001010]に対応する置換ビット列BP2のビットパターンとして、[000]を選択することになる。ビットパターン選択回路15には、例えば図8(e)に示すタイミングによりタイミングジェネレータ16からタイミング信号D108が供給されている。ビットパターン選択回路15は、上記タイミング信号D108のパルス出力タイミングにตอบสนองして、上述のようにして選択した[000]の3ビットパターンを置換ビット列D107として発生してセクタ13に出力する。

【0066】セクタ13には、例えば図8(f)に示すシフトレジスタ12によりシフトされた(1, 7) RLL符号列D103と、上記図8(d)に示す3ビットの置換ビット列D107が入力され、また、図8(g)に示すタイミング信号D109が入力される。タイミング信号D109は、置換ビット列D107の挿入タイミングに対応してHレベルとなる信号である。セクタ13は、タイミング信号D109がLレベルの期間は、図8(f)の(1, 7) RLL符号列D103を出力し、タイミング信号D109がHレベルの期間は図8(d)の置換ビット列D107を出力するように動作する。ただし、タイミング信号D109がHレベルからLレベルに反転した時点から3ビット長に対応する(1, 7) RLL符号D103については、そのデータを捨てるように処理を実行する。これにより、セクタ13からは例えば図8(h)に示す(1, 7) RLL符号列D104が出力される。この(1, 7) RLL符号列D104は、6ビットのDSV制御コードBP1に置き換えて、DSV制御前の(1, 7) RLL符号列を形成する3ビットの置換ビット列BP2を挿入したものである。つまり、DSV制御が解除された(1, 7) RLL符号列である。

【0067】上記(1, 7) RLL符号列D104は、(1, 7) RLLデコード14に供給されて(1, 7) RLL復調処理が施され、出力データD105として出力される。例えば、図8(h)に示す(1, 7) RLL符号列D104であれば、(1, 7) RLLデコード14によって、図8(i)に示すビット列の出力データD105が得られることになる。この出力データは例えばこの後、図示しない信号処理回路に供給されて、所定フォーマットの時系列データとして処理される。

【0068】なお、上記実施の形態においては、被置換ビット列(置換ビット列)BP2を3ビットとし、DSV制御コードBP1を6ビットとした場合について説明したが、例えば4ビットの冗長ビットをDSV制御コードとして単に挿入する場合よりも低い冗長度を得られるのであれば、被置換ビット列(置換ビット列)BP2のビット数と、DSV制御コードBP1のビット数の関係は、任意に設定されて構わない。また、上記実施の形態では、(1, 7) RLL符号に対してDSV制御を行う場合について説明したが、他の(d, k)制限による

(1, 7) RLL符号に対してDSV制御を行う場合にも、当然のこととして本発明を適用することが可能である。さらには、本発明は例えばNRZI変調の代わりにNRZ(Non Return To Zero)変調が採用されることを前提とした場合に対応することも可能である。

#### 【0069】

【発明の効果】以上説明したように、本発明はRLL符号列を形成するaビットの被置換ビット列を、例えば

(d, k)制限を守るための所定規則に従って選択され

たビットパターンを有するm(ただし $m > a$ )ビットのDSV制御コードに置き換え挿入するようにしてDSV制御を行うように構成しているが、これにより、DSV制御が施されたRLL符号として、(d, k)制限を守ったうえで、単にRLL符号列に対して冗長ビットとしてのDSV制御コードを挿入する場合よりも低い冗長度を得ることが可能になり、更に高密度記録化に有利となる。

【0070】また、上記のようにしてDSV制御が施されたRLL符号列についてDSV制御を解除するには、例えば上記DSV制御コードのビットパターンと、DSV制御前の元のデータを形成する置換ビット列(上記被置換ビット列と同等)のビットパターンの対応を示す変換テーブルを用意し、この変換テーブルを参照して選択されたビットパターンの置換ビット列をDSV制御コードと置き換えて挿入するように構成すれば、例えば特に複雑な処理を実行しなくともDSV制御前のRLL符号列を復元することができる。

#### 【図面の簡単な説明】

【図1】本発明の実施の形態のDSV制御コードの概念を示す説明図である。

【図2】本実施の形態においてDSV制御コードの候補となり得るビットパターンを示す説明図である。

【図3】本実施の形態の変換テーブルを示す説明図である。

【図4】同一(1, 7) RLL符号列に対して挿入するDSV制御コードのビットパターンの相違によるNRZI変調符号の波形の相違を示す図である。

【図5】本実施の形態のエンコーダの構成を示すブロック図である。

【図6】エンコーダの信号処理例を示すタイミングチャートである。

【図7】本実施の形態のデコーダの構成を示すブロック図である。

【図8】デコーダの信号処理例を示すタイミングチャートである。図である。

【図9】(1, 7) RLL符号に対するDSV制御方法として、単に冗長ビットとしてのDSV制御コードを挿入する場合を示す説明図である。

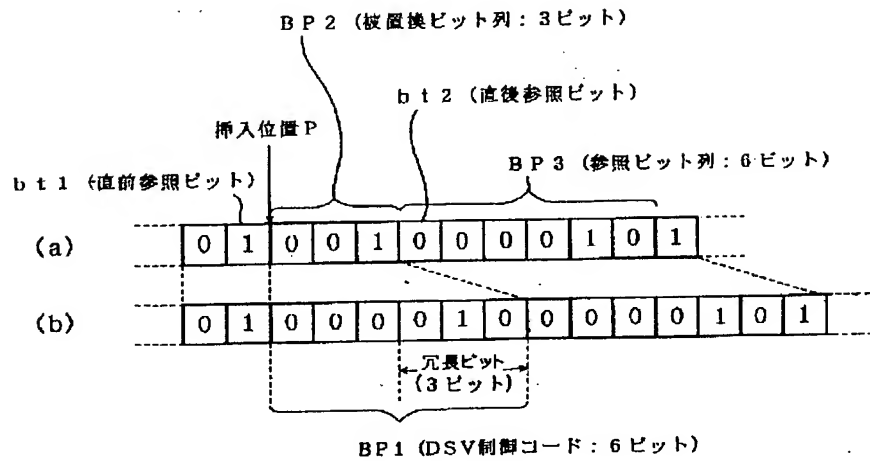
【図10】(1, 7) RLLに対して挿入するDSV制御コードのビット数と、完全DSV制御の可否との関係を示す説明図である。

#### 【符号の説明】

BP1 DSV制御コード、BP2 被置換ビットパターン、BP3 参照ビット列、bt1 直前参照ビット、bt2 直後参照ビット、1 (1, 7) RLLエンコーダ、2, 12 シフトレジスタ、3, 13 セクタ、4 NRZIエンコーダ、5, 15 ビットパターン選択回路、5a, 15a 変換テーブル、6 DSV演算器、7, 16 タイミングジェネレータ、11

NRZ I デコーダ、14 (1, 7) RLL デコーダ

【図1】



【図2】

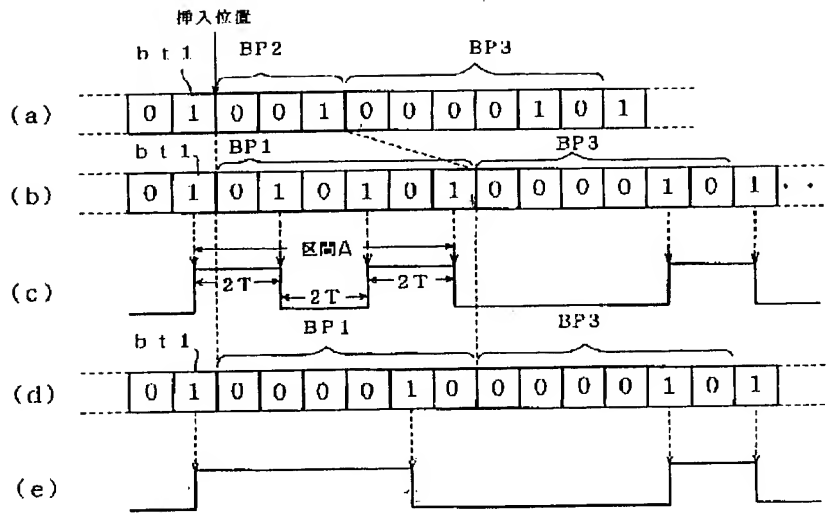
No.	DSV制御コード候補
1	000000
2	000001
3	000010
4	000100
5	000101
6	001000
7	001001
8	001010
9	010000
10	010001
11	010010
12	010100
13	010101
14	100000
15	100001
16	100010
17	100100
18	100101
19	101000
20	101001
21	101010

【図3】

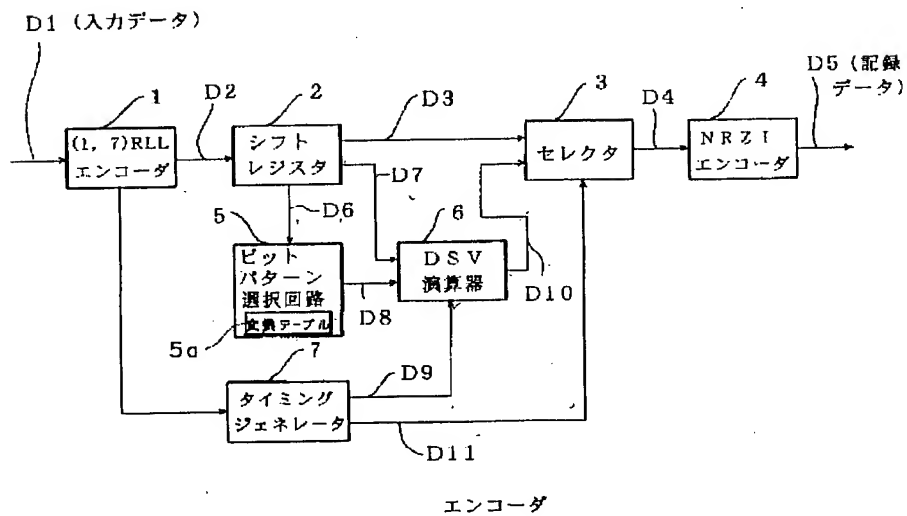
被置換ビット列 (置換ビット列)	DSV制御コード	制御種別	条件
0 0 0	001010	非反転	_____
	000100	反転	_____
0 0 1	001001	非反転	_____
	000010	反転	直前参照ビットが '1' かつ参照ビット列内に 1つ以上 '1' がある
	010101		上記以外
0 1 0	010010	非反転	_____
	100101	反転	直前参照ビットが '0' かつ、直後参照ビットが '0'
	000001		直前参照ビットが '1' かつ、直後参照ビットが '0'
	010000		直後参照ビットが '1'
1 0 0	100100	非反転	_____
	100000	反転	直後参照ビットが '1'
	101010		直後参照ビットが '0'
1 0 1	100001	非反転	_____
	101001	反転	_____

変換テーブル

【図 4】

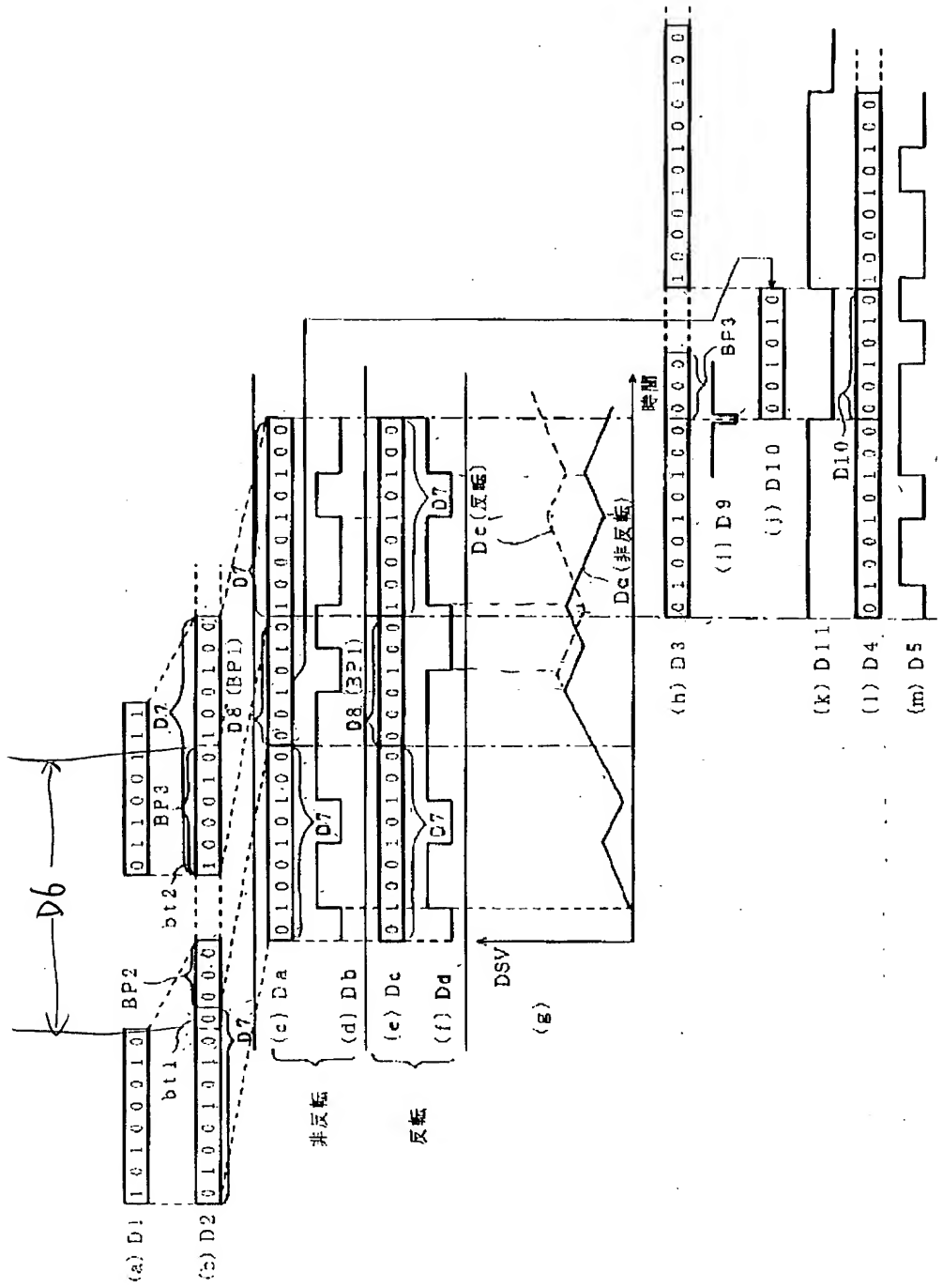


【図 5】

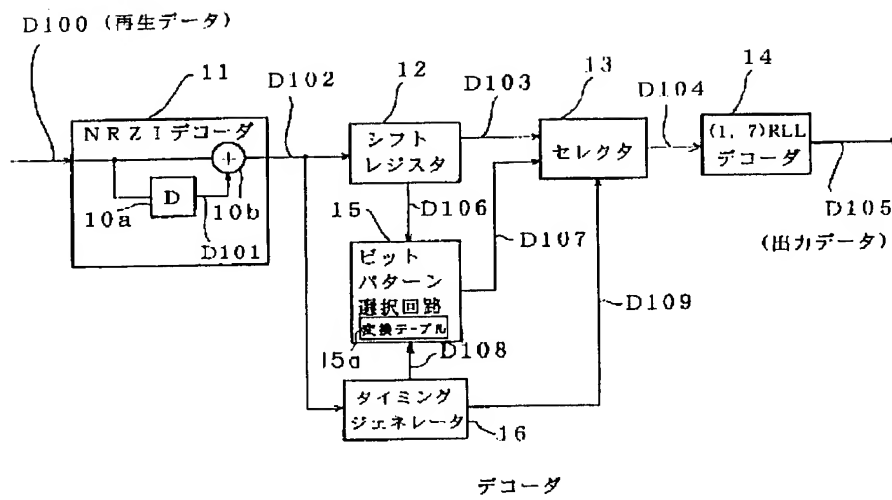




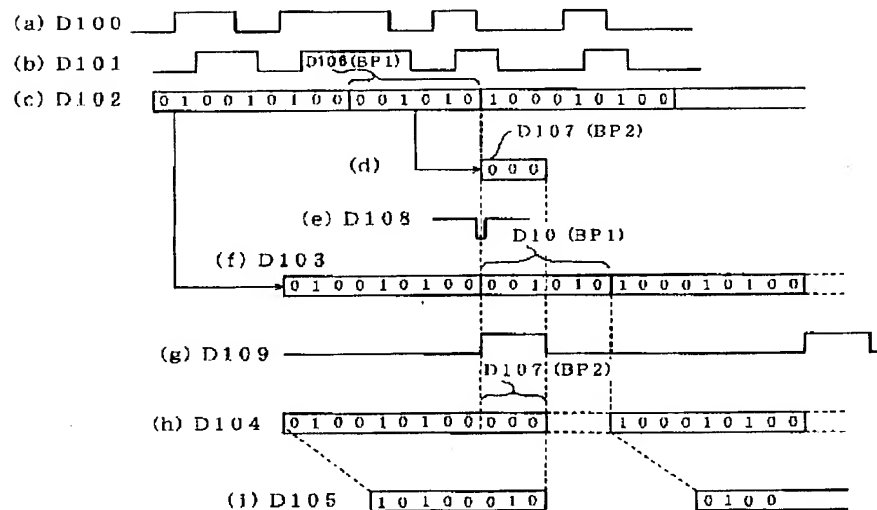
【図6】



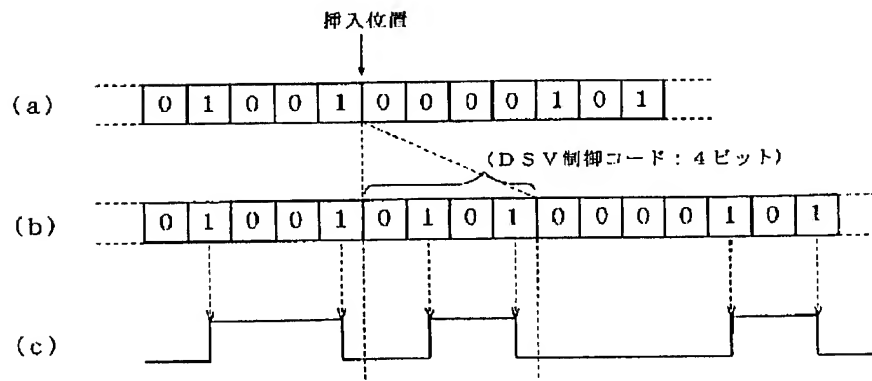
【図7】



【図8】

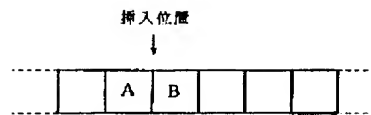


【図9】



【図 1 0】

(a)



(b)

A	B	制御種別	DSV制御コード (2ビット)	(d, k) 制限	DSV制御コード (3ビット)	(d, k) 制限	DSV制御コード (4ビット)	(d, k) 制限
0	0	反転	1 0 (0 1)	○	0 1 0	○	0 1 0 0 (0 0 1 0)	○
		非反転	0 0	×	1 0 1	○	1 0 0 1	○
1	0	反転	0 1	○	0 0 1	○	0 0 0 1	○
		非反転	0 0	×	0 0 0	×	0 1 0 1	○
0	1	反転	1 0	○	1 0 0	○	1 0 0 0	○
		非反転	0 0	×	0 0 0	×	1 0 1 0	○